"The Squawk Bot": Joint Learning of Time Series and Text Data Modalities for Automated Financial Information Filtering (Supplementary Document)

A Thomson Reuters Data

Fig. 1 plots the distribution of numbers of news headlines released per day by Reuters. As a skewed distribution (mode being around 12 headlines per day), we limit all models to explore up to 25 news articles per day. This setting retains more than 95% of the total number of headlines in the original textual corpus. For a small portion of days having exceptionally large numbers of daily news articles (those located to the far-right of the distribution shown in Fig. 1), we keep the last 25 news headlines on each of these days. We use NLTK toolkit (available at `https://www.nltk.org/`) for tex preprocessing, keep the vocabulary size at 5000 unique words, and remove ones that are not found in the 400k words of the GloVe.

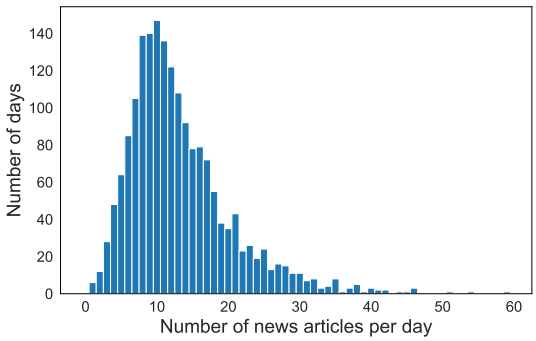

Figure 1: Distribution of numbers of news headlines released per day from the Thomson Reuters news corpus.

B Statistics on Ground Truth News Headlines

Table 1: Statistics on "ground truth" news (GTn) headlines on test dataset associated with company-specific time series. %GTn shows percentage of GTn over all news headlines. %GTd shows percentage of days having at least one GTn. GTn/GTd shows average percentage of GTn on GTd. Max#GTn shows the maximum number of GTn headlines in a single day.

Dataset	%GTn	%GTd	GTn/GTd	Max#GTn
AAPL	8%	45%	15%	9
GOOG	3.5%	26%	12%	5

Submitted to xxx-x-xxxx-xxxx-x/YY/MM. Do not distribute.

C Parameters Setting

We set the parameter ranges for our models and its counterparts as follows: The number of layers' neural units $d_s, d_h \in \{16, 32, 64, 128\}$, regularization $L_1, L_2 \in \{0.1, 0.05, 0.01, 0.005, 0.001\}$, dropout rate $\in \{0.0, 0.1, 0.2, 0.4\}$, and the number of timesteps (sequent length) for stock time series $m \in \{3, 5, 7, 10\}$. We choose 50 as the word embedding dimension with the usage of the pre-trained GloVe model Pennington and others [2014]. For `GRUtxt`, we follows the model presented in Yang and others [2016] using two Bi-LSTMs along with a self-attention layer applied at the sentence level. With CNNtxt, we implement an architecture with three kernel filters of sizes of $\{2, 3, 4\}$, using max pooling as recommended in Kim [2014]; Zhang and Wallace [2015]. We use two layers of LSTMs as encoder-decoder for the `GRUts`, while SVM is used with the linear kernel and C is tuned from the log range Our model and the baselines were implemented and trained on a machine with two Tesla K80 GPUs, running Tensorflow 1.3.0 as backend. We performed the random search using the validation set to tune hyperparameters.

The final values we used for AAPL dataset is $d_s, d_h = 64, m = 5, L_1 = 0.01, L_2 = 0.2$, GOOG is $d_s = 32, d_h = 64, m = 5, L_1 = 0.001, L_2 = 0.1$, both with dropout of 0.2 and using pre-trained GloVe embedding, For S&P500, the values are $d_s = 64, d_h = 96, m = 10, L_1 = 0.001, L_2 = 0.005$ with dropout of 0.1 and initializing with GloVe for word embedding.

D Evaluation Metrics

In empirical evaluation of our models and its counterparts, we have used the precision and recall measurements. While their calculation with respect to predicting class labels remains as usual, their computation with respect to the ground truth news headlines (GTn) is slightly changed, being adaptive to the setting k as the number of top relevant headlines to be returned, as reported in Fig. **??**(a-b), Fig.**??**(a-b). This is because some days have as few as only one GTn, other days may have as many as 9 (in AAPL) or 5 (in GOOG) as shown in Appendix B, while the cardinality k is fixed in computing precision and recall. Specifically, we compute the Pre@k and Rec@k (reported in Fig. **??**(a-b), Fig.**??**(a-b)) averaged from all days having at least one ground truth headline as follows:

$$\text{Pre@}k = \frac{1}{\text{GTd}} \sum_{i=1}^{\text{GTd}} \frac{TP_i[:k]}{TP_i[:k] + FP_i[:k]} \tag{1}$$

$$\text{Rec@}k = \frac{1}{\text{GTd}} \sum_{i=1}^{\text{GTd}} \frac{TP_i[:k]}{\min(k, |\text{GTn}_i[:k]|)} \tag{2}$$

in which $TP_i[:k], FP_i[:k]$ are respectively the numbers of true positive and false positive headlines at day i, and $\text{GTn}_i[:k]$ denotes the number of ground truth news headlines up to k cardinality on day i. The denominator in Rec@k ensures that the number of false negative headlines will not be beyond either the cardinality k or the actual number of ground truth headlines on day i.

E Experiment on S&P500 Time Series

We show in Table 2 the headline news in three days selected from the test dataset, along with the probability mass of attention (on valid news headlines) of our `MSIN` model, its variant `LSTMw/o` and `GRUtxt` respectively in 3rd-1st columns. As observed, our model MSIN places more probability mass on the headlines whose contents directly report the market performance, while zeros out mass on headlines that report company-events or local markets. Its attention mass is also more condense, with clear focus on a small set of top relevant news, as compared to those of `LSTMw/o` and `GRUtxt` that both spread out on multiple headlines. The headlines highlighted in Table 2 are those outputted by our model by setting their cumulated probability mass of at least 50%.

The focus of our work is on discovering relevant text articles in association with the numerical time series. As using the global market time series for experiment, nevertheless, we would like to test the impact of selected relevant text articles discovered by our model in forecasting the overall market movement in the subsequent day. For this task, we further compare its performances against other

Table 2: News headlines from three days selected from the test set, along with the attention mass (annotated in the numeric columns) by: GRUtxt, LSTMw/o and MSIN. Colored headlines are relevant news outputted by our model based on setting the accumulated probability mass $\geq 50\%$.

GRUtxt	LSTMw/o	MSIN	News headlines
			Date: 2013-02-03
0.00	0.53	0.00	01. china service slow uptick highlight recovery
0.08	0.10	0.00	02. japan finance minister weak yen result not goal anti deflation policy
0.02	0.01	0.18	03. france track meet percent growth target minister
0.05	0.16	0.00	04. watch central banker say not
0.12	0.01	0.08	05. us small business borrowing rise december but barely
0.28	0.08	0.07	06. gauge us business investment plan edge low
0.15	0.07	0.39	07. global share euro fall sharply renew euro zone fear
0.30	0.03	0.21	08. sp post bad day since november hill share sink
			Date: 2013-04-26
0.00	0.00	0.00	01. japan growth strategy fiscal plan g8
0.00	0.01	0.00	02. boj credibility test division emerge over inflation target
0.28	0.04	0.01	03. consumer sentiment wane april
0.00	0.01	0.05	04. economic growth gauge year high last week ecri
0.00	0.01	0.00	05. microsoft get upper hand first google patent trial
0.02	0.01	0.01	06. growth fall short forecast weakness ahead
0.22	0.09	0.14	07. dollar fall yen bond yield decline
0.28	0.80	0.73	08. wallstreet dip gdp but finish week high
0.19	0.04	0.05	09. wallstreet week ahead central bank data steer investor
			Date: 2013-10-18
0.00	0.02	0.00	01. china economy show sign slow september stats bureau
0.00	0.01	0.00	02. china share up comfort data but still weekly loss
0.00	0.08	0.01	03. china third quarter gdp growth fast year but outlook dim
0.02	0.26	0.07	04. google third quarter beat ad volume grows stock flirt level
0.01	0.11	0.03	05. schlumberger baker top estimate global drilling
0.26	0.02	0.01	06. us data boon computer driven trading
0.00	0.00	0.00	07. fed release delayed report industrial output october
0.00	0.00	0.00	08. data delay add us inflation bond sale
0.20	0.10	0.07	09. stock split wane may follow google 1,000
0.06	0.03	0.41	10. global stock gain dollar fall fed see stay course
0.44	0.31	0.39	11. sp break record google stock top earnings
0.01	0.03	0.00	12. google share break mobile pay

baseline models, including `CNNtxt` Kim [2014] that both exploit the textual news modality; `GRUts` that analyzes the time series; and `LSTMpar`, closely related to Akita and others [2016], which trains in parallel two LSTM networks, each on one data modality, then fuses them together prior to the output layer. Moreover, for a conventional machine learning technique, we implemented `SVM` Weng *et al.* [2017] that takes in both time series (as vectors) and the uni-gram for the textual news.

Table 3 reports the forecasting accuracy, precision and recall of all models. Two values in each entry of precision and recall columns are respectively reported for the forecasting of up and down market on the next day respectively. As observed, there is not much difference in the prediction accuracy of `GRUtxt` and `CNNtxt` though they analyze the textual news with different network topologies. Without exploiting the temporal order in the time series and the semantic dependencies of textual news, `SVM`'s performance seems less comparable to its counterpart neural networks. Out of all models, our `MSIN`

Table 3: Performance of all models on forecasting market movement. Precision and recall are reported w.r.t. two prediction of up and down market respectively.

Models	Acc.	Pre.	Rec.
LSTMw/o	55.8	**63.9**/48.2	56.1/**56.4**
LSTMpar	53.6	57.8/43.5	70.1/31.3
GRUts	52.8	58.1/43.4	65.7/33.6
GRUtxt	55.3	60.1/45.1	70.9/33.3
CNNtxt	55.9	61.4/46.9	64.1/45.3
SVM	53.1	58.1/41.2	**71.7**/27.4
MSIN	**56.4**	63.3/**50.1**	56.9/56.3

achieves higher prediction accuracy. Its solution also converges to a more balanced classification boundary as reflected in the precision and recall measures.

Table 4: Performance of all models on forecasting market movement as varying the daily latest time up to which news headlines are collected.

Models	9:00	15:40	19:00	23:59
LSTMw/o	56.1	57.2	67.7	77.6
LSTMpar	54.5	56.1	65.9	76.1
GRUtxt	56.9	58.1	68.3	78.2
CNNtxt	56.8	**60.1**	69.1	78.4
SVM	54.3	55.8	61.2	75.9
MSIN	**57.1**	59.8	**69.6**	**79.3**

In order to further observe the impact of textual news on the models' classification accuracy, we vary the latest time, up to which the daily news headlines are collected, to 9:00 (before market open), 15:40 (before market close), 19:00, and 23:59 on the same day the market performance is predicted. These evaluations are reported in Table 4. A clear trend is seen that, the prediction accuracy is higher as the news articles are collected closer to the market closing time, which confirms the indicative information embedded in the textual news toward predicting the S&P500 performance. While most of neural models tend to perform feature engineering better than that of `SVM`, only our `MSIN` can further offer better interpretation due to its join-training of both evolving time series and the news articles. It is noted that evaluations at the time stamp of 19:00 (or 23:59) will shift a model from a predictive system to a purely explanatory one.

References

R. Akita et al. Deep learning for stock prediction using numerical and textual information. In *IEEE/ACIS*, 2016.

Yoon Kim. Convolutional neural networks for sentence classification. *arXiv preprint arXiv:1408.5882*, 2014.

Jeffrey Pennington et al. Glove: Global vectors for word representation. In *EMNLP*, 2014.

Bin Weng, Mohamed A Ahmed, and Fadel M Megahed. Stock market one-day ahead movement prediction using disparate data sources. *Expert Systems with Applications*, 79:153–163, 2017.

Z. Yang et al. Hierarchical attention networks for document classification. In *Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, 2016.

Ye Zhang and Byron Wallace. A sensitivity analysis of (and practitioners' guide to) convolutional neural networks for sentence classification. *arXiv preprint arXiv:1510.03820*, 2015.